Claims

1. A method for encrypting binary data comprising of blocks of tokens, which in turn are comprised of bits, into a binary cipher, comprising the steps of:

segregating a block of binary data from the input stream, making multiple copies of it, and moving the significant digits into the lower bits of the tokens according to a predefined pattern;

modifying the said significant digits by adding their location to their values;

replacing the other (non-significant) binary digits by pseudo-random bits;

rotating segments, which are groups of tokens, of the resulting block by values derived from the count of the bits with a predetermined value of one or zero in the said segments;

modifying the tokens by adding their locations to their values;

rotating the resulting block by a value derived from the count of the bits with a predetermined value of one or zero in the block;

performing a token by token substitution transformation on the block by using a private key, which is a permutation of all possible tokens;

performing a token by token transposition transformation on the block, using a private key, which is the permutation of all possible locations.

2. The system and method as defined in claim **1** wherein the segregation of the blocks is done under the control of two parameters, the *t* token length (number of bits in a token) and the *b* block length (number of tokens in a block).

3. The system and method as defined in claim **2** further comprising the step of inserting one or more authentication tokens into the data at any desired location.

4. The system and method as defined in claim **3** further comprising the step of making a plurality of copies of the data according to parameter c (the number of copies), and thus generating a complete block.

5. The method as defined in claim **4** further comprising a method to change the frequency distribution of the tokens in the said complete block by the following steps:

moving the significant bits of each token to the lowest bits according to a pattern for each copy of the data;

summing the location as a binary number and value as a binary number modulo 2^t for each token and changing the value of the token to this result;

filling the non-significant bits of the tokens with pseudo-random bits;

generating an S_i rotation amount for each segment and rotating it;

summing the location as a binary number and value as a binary number modulo 2^t for each token again;

generating an S_T rotation amount for the complete block and rotating it.

6. The method as defined in claim **5** wherein the pattern for moving the significant bits is a further parameter of the system. This pattern defines which bits are significant in each copy. All combinations work, which satisfy the following criteria: every block has to have at least two significant bits and each source bit has to be represented at least in one copy as significant.

7. The method as defined in claim **5** further comprising a method to generate a count for segment rotation (S_i) by the following steps:

XORing the bits of the bit displacement value into the token displacement value in reverse order;

rotating the count by one bit to the left;

replacing the lowest order bit by the complement of the second lowest order bit.

8. The method as defined in claim **5** further comprising a method to generate a count for complete block rotation (S_T) by the following steps:

XORing the bits of the bit displacement value into the token displacement and segment displacement values in reverse order;

rotating the count by one bit to the left.

9. The system and method as defined in claim **1** further comprising a method to encrypt the data by the following steps in any sequence:

performing a token by token substitution transformation on the modified block by using a private key, which is a permutation of all possible tokens;

performing a token by token transposition transformation on the block resulting from the substitution, using a private key, which is the permutation of all possible locations.

10. The method to mask token frequencies comprising the steps of:

distributing the bits of a token among a plurality of tokens;

moving these bits to the lowest order bits of the tokens;

replacing the other bits with pseudo-random bits;

summing the location as a binary number and value as a binary number modulo 2^t for each token.

11. The method to use the count of bits with a predetermined value of one or zero in a bit string as the rotational value for the string.

12. A method for decrypting binary data from a binary cipher, comprising the steps of:

performing a token by token transposition transformation on the block, using a private key, which is the reversal key of the encryption key;

performing a token by token substitution transformation on the block by using a private key, which is the reversal key of the encryption key;

rotating the resulting block by a value derived from the count of the bits with a value of one in the block;

modifying the tokens by subtracting their locations from their values;

rotating segments of the resulting block by values derived from the count of the bits with a value of one in the said segments;

modifying the tokens by subtracting their locations from their values;

merging the bits from all the copies according to the reversal pattern of the encryption pattern.